



MySQL

Project Idea: Blood Donation Management System

Aim:

The aim of this internship is to provide students with hands-on experience in **database management and backend development** by designing and implementing a **Blood Donation Management System** using **MySQL**. This project will help students understand how to **manage donor and recipient data, process blood donation requests, and track blood inventory efficiently**.

Objectives:

1. Understanding MySQL Database Management:

- Learn the fundamentals of relational databases and SQL.
- Understand database normalization, indexing, and query optimization.

2. Blood Donation & Inventory Management:

- Create a database to store donor and recipient details.
- Implement a blood bank inventory to track blood units by blood group.
- Ensure real-time updates of blood stock when donations and transfusions occur.

3. Donor & Recipient Registration:

- Implement tables for donor registration (Name, Age, Blood Group, Contact, Last Donation Date).
- Create tables for recipients (Name, Age, Blood Group, Contact, Blood Required).

4. Donation & Request Processing:

- Store donation records linked to donors and blood bank inventory.
- Allow recipients to request blood units, check availability, and approve/reject requests.
- Prevent blood requests if stock is insufficient.

5. Order Tracking & Reports:

- Generate reports on blood donations and transfusions.
- Track blood availability for different blood groups.
- Provide statistics on donor participation and stock levels.



ARTTIFAI TECH

Expected Output:

At the end of this internship, students will develop a **fully functional MySQL-based Blood Donation Management System** with:

- ✓ A structured **database schema** with **tables for donors, recipients, donations, blood inventory, and requests.**
- ✓ A system to **register donors and recipients dynamically.**
- ✓ The ability to **track blood availability and prevent out-of-stock requests.**
- ✓ **Automated stock updates** when donations occur.
- ✓ **Reports and analytics** on **donations, transfusions, and stock levels.**

Sample Output:

- **Donor Registration**

Input SQL Query:

```
INSERT INTO Donors (Name, Age, BloodGroup, Contact, LastDonationDate)
VALUES ('Rahul Sharma', 28, 'O+', '9876543210', '2025-03-01');
```

Output:

New donor registered: Rahul Sharma (O+)

- **Recipient Registration**

Input SQL Query:

```
INSERT INTO Recipients (Name, Age, BloodGroup, Contact, BloodRequired)
VALUES ('Aditi Verma', 35, 'O+', '9898989898', 2);
```

Output:

New blood request registered: Aditi Verma (O+) needs 2 units.

- **Blood Donation Entry**

Input SQL Query:

```
INSERT INTO Donations (DonorID, BloodGroup, DonationDate)
VALUES (1, 'O+', '2025-03-20');
```

Trigger Execution (Automatic Stock Update):

```
UPDATE BloodInventory
SET UnitsAvailable = UnitsAvailable + 1
WHERE BloodGroup = 'O+';
```

Output:

Donation recorded: Donor ID 1 (O+) donated 1 unit.

Updated Blood Inventory: O+ → 6 units available.

- **Blood Request Processing**

Recipient requests 2 units of blood (O+). Input SQL Query:

```
SELECT UnitsAvailable FROM BloodInventory WHERE BloodGroup = 'O+';
```

Output:

Available stock for O+: 6 units.



ARTTIFAI TECH

- **Blood Inventory Report**

Input SQL Query:

SELECT * FROM BloodInventory;

Output:

Blood Group	Units Available
O+	4
A+	5
B+	3
AB+	6
O-	2
A-	1

- **Donation History Report**

Input SQL Query:

SELECT Donors.Name, Donations.BloodGroup, Donations.DonationDate
FROM Donations

JOIN Donors ON Donations.DonorID = Donors.DonorID;

Output:

Name	Blood Group	Donation Date
Rahul Sharma	O+	2025-03-20
Pooja Nair	A+	2025-03-15

Note:

- The completed project must be uploaded to the student's own GitHub repository.
- The GitHub repository must be public for evaluation purposes.
- The repository link should be submitted via the Google Form:
<https://forms.gle/tNFMkX5wt343vpCo7>